CHAPTER 10

BY:

VISHU GOYAL

ETISHA JAIN

LINUX 10.1 Understanding RPM Packages

- RPM stands for Redhat Package Manager.
- Red Hat package Manager is a package format to install software, as well as a database of installed packages on a system.
- The package contains on archive of files that is compressed with cpio, as well as metadata and a list of package dependencies
- RPM Packages may contain scripts as well which helps us to set or automatically download all dependencies
- to install packages, repositories are used, it will helps us a lot becox they will install all dependability
- Individual Packages may be installed but this should be avoided. Using repositories is helpful as you don't only have just the software packages, but also all the libraries and dependencies that exist for these software packages

LINUX 10.2 Setting up Repository Access

- We will create a **local repository** so that we can install packages from RHEL 8 installation disk ISO image.
- Create an ISO image:
  dd if=/dev/sr0 of=/rhel8.iso bs=1M
- Create a directory /repo: mkdir /repo
- Edit **/etc/fstab** and add the following line to the end
  Vim /etc/fstab
  /rhel8.iso /repo iso9660 defaults 0 0
  We need to run systemctl daemon-reload
- Use **mount -a** to mount the ISO
- cd ISO BaseOS
- repodata is the YUM repository Metadata.
- createrepo to generate repodata ourself.
  We can use this directory in the repository client.

- **Create the file/etc/yum.repos.d/appstream.repo with following contents**

[appstream] I.e. vim appstream.repo
name=appstream
baseurl=file:///repo/AppStream
gpgcheck=0

Two - one for Base Packages ,one for AppStream

- **Create the file/etc/yum.repos.d/base.repo with following contents**

[base] I.e. vim base.repo
name=base
baseurl=file:///repo/BaseOS
gpgcheck=0

LINUX 10.3 Understanding Modules and Application Streams

- RHEL8 introduces application streams and modules ie. more efficient way to work with softwares.
- Application streams are used to separate user space packages from core kernel operations.(multiple versions of applications can be and applications can be updated independently).
- Using application streams allows for working with different versions of packages
- Base packages are provided through the BaseOS repository.
- AppStream is provided as a separate repository.
- Application Streams are delivered in two formats
- Traditional RPMs.
- New Modules.
- Modules can contain streams to make multiple versions of applications available.
- Enabling a module stream gives access to RPM Packages in that stream
- Modules can also have profiles. A profile is a list of packages that belong to a specific use-case.(minimal installation, complete installation, recommended installation).
- The package list of a module can contain packages outside the module stream.
- Use the YUM module commands to manage Modules.

LINUX 10.4 Managing Packages with yum

- **yum** was created to be intuitve
- **yum search nmap**
- **yum install nmap**
- **yum remove vi** it can't be removed easily becox they depend on dependancies
  it is generating error becox the operation would result in removing the protected packages:
  sudo . protected packages can not removed easily unless they are not unprotected.
- **yum remove nmap**
- **yum update**
- **yum update kernel**
- **yum provides** it will search more deep then the yum search
- **yum provides**(closely related to yum search)
- **Yum search** is searching only in package name and description, yum provides is going deeper in the package to see of there are any files that match
- **yum search sepolicy**
- **yum provides */sepolicy**
- **yum info**(info about a package)
- **yum info nmap**
- **yum list** (info about different packages)
- **yum list all** (for packages available and installed packages)
- **yum list installed**

LINUX 10.5 Managing Modules and Application Streams

The yum module command is used to manage module properties

- **yum module list**
- **yum module provides httpd** searches th emodule that provide a specific packages
- **yum module info php**
- **yum module info --profile php** shows profiles
- **yum module list php** show which streams are available
- **yum module install php:7.1 or yum install @pgp:7.1**
- **yum module install php:7.1/devel** installs a specific profile
- **yum install httpd** will have yum automatically enable the module stream this package is in before installing this package
- **yum module enable php:7.1** enables the module but doesn't install anything yet
- **yum module install php:7.1** will install a specific PHP module stream
- This will also enable the 7.1 stream
- **yum module install php:7.2** will update to the newer version
- the 7.1 stream will be disabled, and the 7.2 stream will be enabled.
- To update or downgrade packages from a previous module stream that are not listed in the profiles that are installed with the module update, use **yum distro-sync**
- **yum module list** to show all the module
- - yum module list
- - yum module info php
- - yum module info --profile php:7.1
- - yum module instal php:7.1
- - yum update
- - yum module install php:7.2
- - yum module install php:7.1

LINUX 10.6 Using yum groups

- **yum groups** are provided to give access to specific categories of spftware
- **yum groups list** gives a list of most common yum groups
- **yum groups list hidden** shows all yum groups.
- **yum groups info <groupname>** shows which packages are in a group.
- **yum groups install <groupname>** will install a specific yum group.
- **yum groups install --with-optional "Directory client"**

LINUX 10.7 yum history and yum update

- **yum history** gives a list of recently issued commands
- **yum history** undo allows you to undo a specific command based on the history information
- **yum update** will update all packages on your system
- **yum update <packagename>** will update one package only, including its dependencies

LINUX 10.8 Using RPM Queries

- **rpm** is the legacy command to manage RPM Packages.
- Do not use rpm to install packages as it doesn't consider dependencies **(rpm -i)**
- rpm is **useful** though to perform package queries.
- rpm queries by default are against the database of installed packages, add -p to query

package files.

- **rpm -qf /any/file** --> package name that is the file is coming from.
- **rpm -ql mypackage** --> shows files in a package
- **rpm -qc mypackage** --> shows configuration files
- **rpm -qp --scripts mypackage-file.rpm** ---> any scripts in the file you are going to install.
- **rpm -qp** to query packages that you have in a file and not yet installed

Practical

- **cd /etc/**
- **ls**
- **rpm -qf /etc/tcsd.conf** It tells us from which file the package comes from
- **rpm -ql trousers** shows all the files
- **rpm -qd trousers** show documentation only
- **rpm -qc trousers** show configuration file only
- **man tcsd**
- to home directory
- yum install **dnf-utils -y** (default is package installer of fedora)
- In RHEL 8 call it **yum2 instead of dnf.**
- **DNf** util is giving utility of **yumdownloader.**
- **yumdownloader httpd** it will download the package

- **ls**
- **rpm -qp** (p for package file instead of package which is already installed)
- **rpm -qp --scripts httpd.**

LINUX 10.9 Using Red Hat Subscription Manager

- To work with **RHEL repositories**, you need a subscription
- If you just want to evaluate, use the free developer subscription
  get from https://developer.redhat.com
- Use **subscription manage**r to set up the subscription
- Use **subscription-manager register** to register
  Username(Of redhat)
  Password(Of redhat)
- Use **subscription-manager attach --auto** to connect your current subscription

Subscription doesn't work on **CenteOS**

- yum repolist

THANK YOU